

# Języki i Technologie Webowe

## Protokół HTTP, Przeglądarki

Igor Wojnicki

Katedra Informatyki Stosowanej  
Akademia Górniczo-Hutnicza w Krakowie

16 października 2013

# Spis Treści

- 1 Protokół HTTP
- 2 Klient HTTP: Przeglądarka
- 3 Wyszukiwarki

I. Wojnicki, JiTW

1 Protokół HTTP

2 Klient HTTP: Przeglądarka

3 Wyszukiwarki

I. Wojnicki, JiTW

# Wybrane Definicje HTTP, RFC1945 I

<http://www.ietf.org/rfc/rfc1945.txt>

HTTP Made Really Easy:

<http://www.jmarshall.com/easy/http/>

- połączenie (connection) – komunikacja ustanowiona pomiędzy dwoma aplikacjami,
- komunikat (message) – podstawowa jednostka komunikacji HTTP, na komunikat składają się zdefiniowane zbiory **bajtów**,
- żądanie (request) – komunikat HTTP będący żądaniem,
- odpowiedź (response) – komunikat HTTP będący odpowiedzią,
- zasób (resource) – dane albo usługa zidentyfikowana przez URI,
- dane (entity) – reprezentacja zasobu,
- klient (client) – aplikacja nawiązująca połączenie celem wysyłania żądań,
- serwer (server) – aplikacja przyjmująca połączenia, wysyła odpowiedzi.

# Podstawy HTTP

- Hyper Text Transfer Protocol.
- HTTP 1.0, 1.1.
- Wykorzystuje TCP, domyślnie port 80.
- Typ komunikacji: Żądanie-Odpowiedź.
- Specyfikacja zasobu poprzez podanie URI, obsługiwane schematy URI: `http`, `https`.
- Do identyfikacji przesyłanych danych używany jest podzbiór MIME (Multipurpose Internet Mail Extensions, RFC1521).

# Struktura Protokołu HTTP

## Żądanie:

- polecenie
- nagłówki (0 lub więcej)
- pusta linia
- dane

## Odpowiedź:

- pojedyncza linia statusu: `protokół kod opis`

# Protokół HTTP: Status

1xx powiadomienie

2xx powodzenie – 200 OK

3xx przekierowanie do innego URI – 301 Moved  
Permanently

4xx błąd po stronie klienta – 404 Not Found

5xx błąd po stronie serwera – 500 Server Error

# Protokół HTTP: Polecenia I

- Wybrane polecenia protokołu (uwaga: rozróżniane duże i małe litery):
  - GET: żądanie udostępnienia zasobu,
  - HEAD: j.w. tylko nagłówek,
  - POST: wysłanie danych do serwera (np. dane z formularzy),



# Protokół HTTP: Nagłówki

Klient:

**From:** – zwykle email

**User-Agent:** – identyfikacja klienta: Nazwa/Wersja

**Host:** – HTTP 1.1

Serwer:

**Server:** – analogicznie jak User-Agent

**Last-Modified:** – Data i godzina modyfikacji zasobu

**Connetion:** – rodzaj połączenia (`close` dla HTTP 1.0)

# Protokół HTTP: Nagłówki, identyfikacja danych

**Content-Type:** – tym MIME, np. `text/html`,  
`application/octet-stream`

**Content-Length:** – rozmiar w bajtach

I. Wojnicki, IITW

# Przykład: HEAD I

```
telnet charon.ia.agh.edu.pl 80
Trying 149.156.199.59...
Connected to charon.ia.agh.edu.pl.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Tue, 28 Feb 2006 13:34:42 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-16 mod_
Last-Modified: Thu, 15 Sep 2005 10:04:15 GMT
ETag: "9a-b81-9978c5c0"
Accept-Ranges: bytes
Content-Length: 2945
Connection: close
Content-Type: text/html
```

# Przykład: HEAD I

```
telnet www.agh.edu.pl 80
Trying 149.156.96.1...
Connected to web.uci.agh.edu.pl.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Tue, 28 Feb 2006 13:36:53 GMT
Server: Apache/1.3.33 (Unix) mod_ssl/2.8.24 OpenSSL/0.9.7d
X-Powered-By: PHP/5.0.4
Connection: close
Content-Type: text/html; charset=iso-8859-2
```

# Przykład: GET I

```
telnet www.agh.edu.pl 80
Trying 149.156.96.1...
Connected to web.uci.agh.edu.pl.
Escape character is '^]'.
GET / HTTP/1.0

HTTP/1.1 200 OK
Date: Tue, 28 Feb 2006 12:57:17 GMT
Server: Apache/1.3.33 (Unix) mod_ssl/2.8.24 OpenSSL/0.9.7d
X-Powered-By: PHP/5.0.4
Connection: close
Content-Type: text/html; charset=iso-8859-2

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
.
.
</html>
```

# Przykład: GET I

```
telnet charon.ia.agh.edu.pl 80
Trying 149.156.199.59...
Connected to charon.ia.agh.edu.pl.
Escape character is '^]'.
GET / HTTP/1.0

HTTP/1.1 200 OK
Date: Tue, 28 Feb 2006 13:00:42 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-16 mod_
Last-Modified: Thu, 15 Sep 2005 10:04:15 GMT
ETag: "9a-b81-9978c5c0"
Accept-Ranges: bytes
Content-Length: 2945
Connection: close
Content-Type: text/html
```

# Przykład: GET II

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">  
<HTML>  
.  
.  
.  
</HTML>
```

I. Wojnicki, JTW

# Przykład: GET, niepowodzenie I

```
telnet charon.ia.agh.edu.pl 80
Trying 149.156.199.59...
Connected to charon.ia.agh.edu.pl.
Escape character is '^]'.
GET /sth/i.html HTTP/1.0
```

```
HTTP/1.1 404 Not Found
Date: Fri, 10 Mar 2006 11:47:24 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-16 mod_
Content-Length: 348
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
```



# Przykład: GET, niepowodzenie II

```
</head><body>
<h1>Not Found</h1>
<p>The requested URL /sth/i.html was not found on this serv
<hr>
<address>Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-16 mod
</body></html>
```

I. Wojnicki, JitW

# Przykład, HEAD, HTTP/1.1 I

- Może być więcej niż jedno polecenie/połączenie.
- Przekazanie nazwy symbolicznej węzła (host), którego dotyczy żądanie, jako część składowa protokołu.

```
telnet charon.ia.agh.edu.pl 80
Trying 149.156.199.59...
Connected to charon.ia.agh.edu.pl.
Escape character is '^]'.
HEAD / HTTP/1.1
Host: charon.ia.agh.edu.pl
```

```
HTTP/1.1 200 OK
Date: Wed, 01 Mar 2006 14:51:09 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-16 mod_
Last-Modified: Thu, 15 Sep 2005 10:04:15 GMT
ETag: "9a-b81-9978c5c0"
```

# Przykład, HEAD, HTTP/1.1 II

```
Accept-Ranges: bytes  
Content-Length: 2945  
Content-Type: text/html
```

I. Wojnicki, JiTW

# Przykład, POST I

```
POST /path/script.cgi HTTP/1.0
From: frog@jmarshall.com
User-Agent: HTTPTool/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 32
```

```
home=Cosby&favorite+flavor=flies
```

# application/x-www-form-urlencoded

- Znaki inne niz litery/cyfry jako %xx, gdzie xx jest wartością ASCII.
- Spacja: +
- nazwa=wartość, oddzielone &
- np. imiona=Igor+Witold&nazwisko=Wojnicki

# GET raz jeszcze

- Stosując GET można również przekazywać parametry podobnie jak dla POST.
- Protokół HTTP nie określa max. długości URI – Istnieje ograniczenie narzucone przez przeglądarki, bezpiecznie 2kB.
- Znaki winny być zakodowane tak jak dla:  
`application/x-www-form-urlencoded`
- `http://www.google.pl/search?q=Protokol+HTTP`

1 Protokół HTTP

2 Klient HTTP: Przeglądarka

3 Wyszukiwarki

I. Wojnicki, JiTW

# Zadania przeglądarki

- Używa HTTP.
- Renderuje HTML, XML, CSS + grafika rastrowa (JPG, PNG, GIF), grafika wektorowa SVG (Scalable Vector Graphics).
- JavaScript.
- System wtyczek:
  - Java.
  - Flash.
- Standardy HTML, XML, CSS: World Wide Web Consortium (W3C): <http://www.w3.org>



# Przeglądarki

- Mozilla/Firefox (Gecko).
- Safari (WebKit: WebCore (KHTML) + JavaScriptCore (kjs (KDE))
- Konqueror (KHTML + kjs)
- Google Chrome.
- Internet Explorer.
- Opera.
- Testy Acid1, Acid2, Acid3: <http://www.acidtests.org/>

# Proxy

**Proxy serwer** system komputerowy pośredniczący w nawiązywaniu połączeń, często używane do HTTP, FTP, HTTPS.

Główne zadania:

- filtracja (ceznura, wirusy, przeciwdziałanie cenzurze, konwersja protokołów),
- buforowanie (szybszy dostęp do danych),
- anonimizacja.

Rodzaje:

- jawny,
- transparentny.

# Proxy i Przeglądarki

- Transparentny – żadna konfiguracja nie jest wymagana.
- Jawny:
  - Ręczne: HTTP, HTTPS, FTP, dla każdego # portu (zwykle 8080).
  - Automatyczne: URI ze *skryptem automatycznej konfiguracji Proxy*.

# Przeglądarka: Automatyyczna Konfiguracja Proxy – Skrypt

Funkcja JavaScript uruchamiana przez przeglądarkę przy odwołaniu do URI:

```
function FindProxyForURL(url, host)
{
    ...
}
```

Zwraca:

- `DIRECT` połączenie bezpośrednie,
- `PROXY host:port` połączenie przez zdefiniowany serwer.

Może zwracać wiele wartości – używane przez przeglądarkę w kolejności, o ile poprzednik nie odpowiada.

# Przykład AKP

Skrypt pod adresem: `http://www.agh.edu.pl/agh.pac`

Zawartość:

```
function FindProxyForURL(url, host)
{
    return "PROXY w3cache.agh.edu.pl:8080";
    return "DIRECT";
}
```

1 Protokół HTTP

2 Klient HTTP: Przeglądarka

**3 Wyszukiwarki**

I. Wojnicki, JiTW

# Wyszukiwarki I

Historycznie:

- Katalogi.
- Indeksy.

Aktualnie rozwiązania hybrydowe.

**Altavista** <http://www.altavista.com>, pierwsza wyszukiwarka,

**Google** <http://www.google.com>,

**Yahoo** <http://www.yahoo.com>,

**yippy** (Vivisimo, Clusty) <http://yippy.com>, klasteryzacja rezultatów.

**mozDex** <http://www.mozdex.com/>, algorytmy opensource,

**dmoz** <http://dmoz.org/>, Open Directory Project, katalog!

**hakia** <http://hakia.com/>, semantyczna,

**WolframAlpha** <http://www.wolframalpha.com> Mathematica, wiedza, obliczenia.

# Google

- Potentat na rynku.
- Unikalne serwisy.  
Użycie AJAX (Asynchronous JavaScript And XML) na szeroką skalę.
  - Mail: <http://mail.google.com> – 7+GB.
  - Maps: <http://maps.google.com> – rewolucyjne zastosowanie AJAX do interakcji (przewijanie mapy); połączenie map wektorowych i satelitarnych, szukanie połączeń,
  - Video: <http://video.google.com> – baza danych materiałów wideo, również komercyjnych, AJAX+Flash;  
<http://youtube.com>.
  - Calendar: <http://www.google.com/calendar>.
  - Docs: <http://docs.google.com>.



# Google Map

Help



Web Images Groups News Froogle Local **more »**

1536 marbella, 63138

Search

Search the map

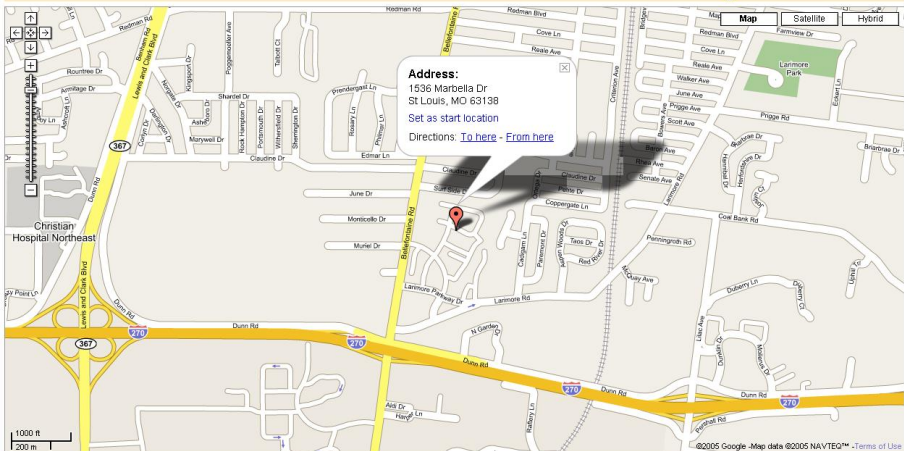
Find businesses

Get Directions

e.g., "hotels near lax;" "10 market st, san francisco"

## Local

Print Email Link to this page



# Google Map

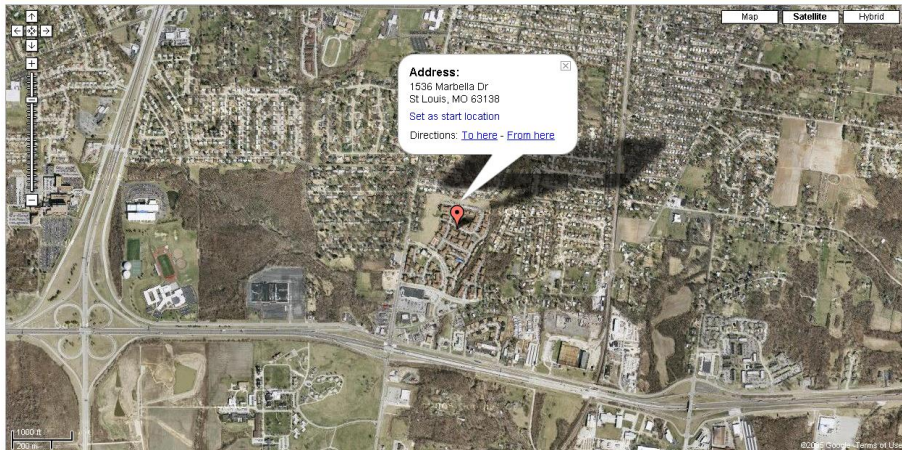
[Help](#)

[Web](#) [Images](#) [Groups](#) [News](#) [Froogle](#) [Local](#) [more »](#)



Search the map  
Find businesses  
Get Directions

e.g., "hotels near lax" or "10 market st, san francisco"

**Local**
[Print](#) [Email](#) [Link to this page](#)


# Google Map

Help



Web Images Groups News Froogle Local **more »**

1536 marbella, 63138

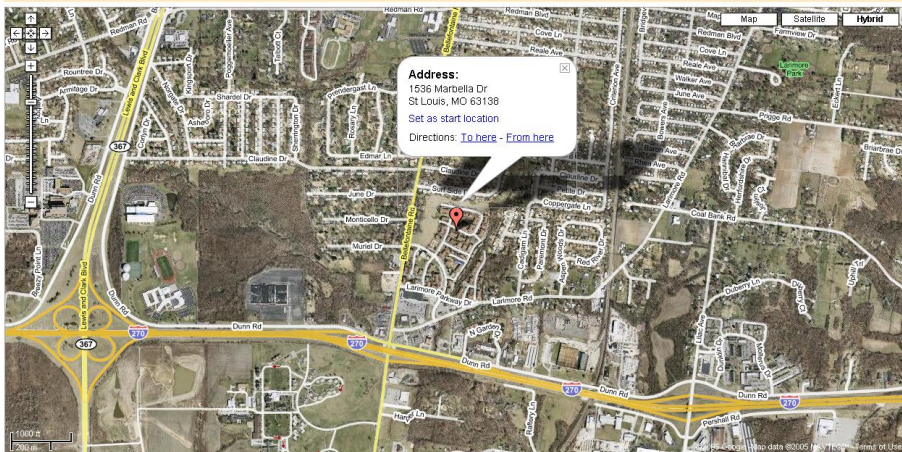
Search

Search the map  
Find businesses  
Get Directions

e.g., "hotels near lax" or "10 market st, san francisco"

## Local

Print Email Link to this page



# Google Map

Help



Web Images Groups News Froogle Local **more »**

1536 marbella, 63138

Search

Search the map

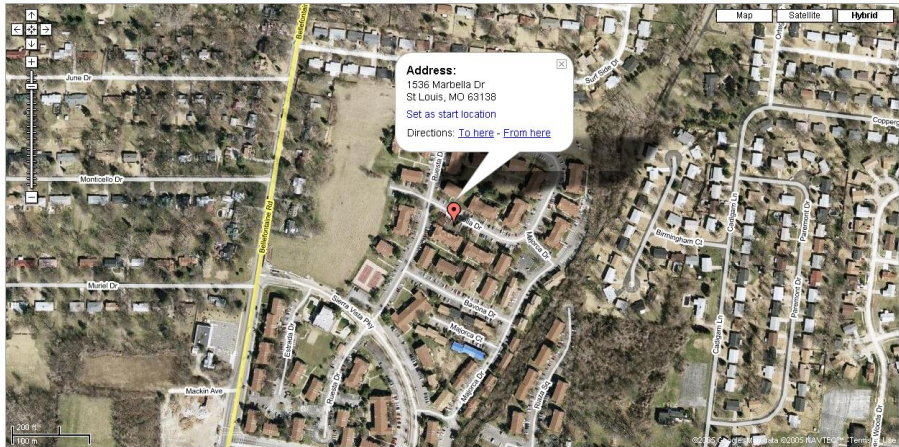
Find businesses

Get Directions

e.g., "hotels near las;" or "10 market st, san francisco"

## Local

Print Email Link to this page





## Google Map

Help


[Web](#) [Images](#) [Groups](#) [News](#) [Froogle](#) [Local](#) [more »](#)

1536 marbella, 63138

Start address:

Natural bridge road, 631

End address:

Search

Search the map  
Find businesses

Get Directions

## Local

Start address: 1536 Marbella Dr  
St Louis, MO 63138

End address: Natural Bridge Rd  
St Louis, MO 63121

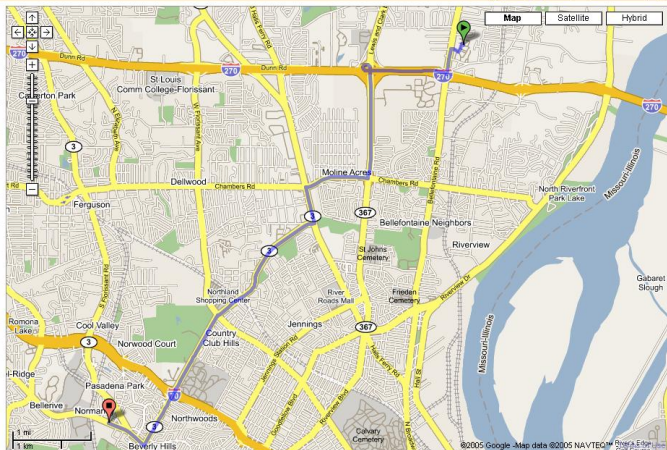
Distance: 8.7 mi (about 17 mins)

## Reverse directions

1. Head **northwest** from **Marbella Dr** - go **130 ft**
2. Turn **left** at **Ruesta Dr** - go **0.1 mi**
3. Turn **right** at **Sierra Vista Pky** - go **0.1 mi**
4. Turn **left** at **Bellefontaine Rd** - go **0.3 mi**
5. Turn **right** at **Dunn Rd** - go **0.1 mi**
6. Bear **left** into the **I-270 W** entry ramp - go **0.9 mi**
7. Take the **MO-367 S** exit **31A** to **St Louis** - go **0.2 mi**
8. Bear **right** at **Lewis and Clark Blvd** - go **1.3 mi**
9. Turn **right** onto the **Chambers Rd West** ramp - go **0.1 mi**
10. Turn **right** at **Chambers Rd** - go **0.8 mi**
11. Turn **left** at **Halls Ferry Rd** - go **0.5 mi**
12. Turn **right** at **Lucas-Hunt Rd** - go **3.7 mi**
13. Turn **right** at **Natural Bridge Rd** - go **0.5 mi**

These directions are for planning purposes only. You may find that construction projects, traffic, or other events may cause road conditions to differ from the map results.

Map data ©2005 NAVTEQ™, Tele Atlas


[Print](#) [Email](#) [Link to this page](#)
 Map  Satellite  Hybrid